

**Konkatenation (Verkettung):** Hintereinanderschreiben von Zeichen

- Das Zeichen  $\cdot$  notiert diese Operation explizit

**Beispiel:**

- $3 \cdot 7 = 37$
- $l \cdot n \cdot f \cdot o \cdot r \cdot m \cdot a \cdot t \cdot i \cdot k = \text{Informatik}$

**Verallgemeinerung:** Auch Zeichenreihen können verkettet werden:

- $(\text{Infor}) \cdot (\text{matik}) = \text{Infor} \cdot \text{matik} =$

$l \cdot n \cdot f \cdot o \cdot r \cdot m \cdot a \cdot t \cdot i \cdot k = \text{Informatik}$

*Infor.matik*

- die Klammern (...) sind sogenannte **Metazeichen** und gehören ebenso wie der Operator  $\cdot$  nicht zu den zu verkettenden Zeichenreihen



**Halbgruppe:** *Semigroup*

Menge  $\mathcal{U}$  mit binärer Operation  $\cdot$ , so daß für alle  $a, b, c \in \mathcal{U}$  gilt:

- $a \cdot b \in \mathcal{U}$  ( *algebraische Abgeschlossenheit* )
- $(a \cdot b) \cdot c = a \cdot (b \cdot c)$  ( Assoziativgesetz ) (HG1)

*f. alle  $a \in \mathcal{U}$   
gibt es  $a'$   
 $e = a \cdot a'$*

**kommutative oder abelsche Halbgruppe:**

- Es gilt zusätzlich für alle  $a, b \in \mathcal{U}$  :
  - $a \cdot b = b \cdot a$  ( Kommutativgesetz ) (HG2)
- Die meisten Halbgruppen in der Informatik sind nicht kommutativ

**Algebra:** Tripel  $\mathcal{A} = (A, O, G)$  wobei

- $A$  ist Menge ( *Trägermenge* )
- $O$  ist Menge algebraisch abgeschlossener Operationen  $\mathcal{U}, \dots$
- $G$  ist Menge von Gesetzen

**Beispiel:** Halbgruppe  $H = (\mathcal{U}, \{\cdot\}, \{HG1\})$  der Zeichenreihen über dem Zeichenvorrat  $\mathcal{U}$



### Beispiele:

- natürliche Zahlen  $\mathbb{N}$  mit Addition  $+$
- natürliche Zahlen  $\mathbb{N}$  mit Multiplikation  $\cdot$
- Mengen mit Vereinigung  $\cup$
- Mengen mit Durchschnitt  $\cap$

$$\mathbb{N} = \{ \underline{0}, 1, 2, \dots \}$$

$$\mathbb{N}_+ = \{ 1, 2, 3, \dots \}$$

- alle diese Halbgruppen sind abelsch (kommutativ)  $a \cdot b = b \cdot a$
- die Halbgruppe der Zeichenreihen mit  $\cdot$  zur Verkettung ist nicht abelsch!
- jede Gruppe i.S.d. Mathematik ist auch eine Halbgruppe
- endliche Halbgruppen werden über Verknüpfungstabellen definiert

$\cdot$	0	1	viele
0	0	1	viele
1	1	viele	viele
viele	viele	viele	viele

$$\mathcal{U} = \{ 0, 1, \text{viele} \}$$



**Monoid:** Halbgruppe  $(U, \cdot)$  mit Element  $\underline{\varepsilon} \in U$ , so daß für alle  $a \in U$  gilt:

- $\varepsilon \cdot a = a$  und  $a \cdot \varepsilon = a$

( Einselement, neutrales Element oder kurz Eins )

**Beispiele:**

- leerer Text in der Halbgruppe der Zeichen
- leere Menge  $\emptyset$  bei Mengen mit Vereinigung  $\cup$
- 0 in der additiven Halbgruppe der ganzen Zahlen
- 1 in der multiplikativen Halbgruppe der ganzen Zahlen

$$\begin{aligned} \text{"abc"} \cdot \text{"def"} &= \text{"abcdef"} \\ \text{""} \cdot \text{"abc"} &= \text{"abc"} \end{aligned}$$



**Linkseins :**  $\underline{1_l \cdot a = a}$  für alle  $a \in U$

**Rechtseins :**  $\underline{a \cdot 1_r = a}$  für alle  $a \in U$

**Beispiel:** Betrachte Halbgruppe

$$U = \{0, 1\}$$

·	0	1
0	0	1
1	0	1

- 0 und 1 sind Linkseinsen
- aber:  $x \cdot 0 = 0$  und  $x \cdot 1 = 1$
- Also ist weder 0 noch 1 Rechtseins

$$0 \cdot 0 = 0$$

$$1 \cdot 1 = 1$$

$$0 \cdot 1 = 0$$

$$1 \cdot 0 = 0$$

Ein Einselement ist eindeutig bestimmt

**Beweis:**

Angenommen es gibt zwei Einsen e und e', dann gilt:

$$e = e' \cdot e \quad \text{weil } e' \text{ Rechtseins ist}$$

$$= e' \quad \text{weil } e \text{ Linkseins ist}$$



## 2.1 Beispiel: Relationen mit Verknüpfung

9

Relationen zwischen Mengen  $U$  und  $V$ :  $\rho \subseteq U \times V$

$$\rho = \{ (u, v) \mid (u, v) \in U \times V \}$$

Verknüpfung zweier Relationen  $\rho \subseteq U \times V$  und  $\sigma \subseteq V \times W$ :

$$\rho \cdot \sigma = \{ (u, w) \mid \exists v \in V: (u, v) \in \rho \wedge (v, w) \in \sigma \} \subseteq U \times W$$

Speziellfall  
zu jedem  $u$   
gibt es ein  $v$   
mit  $u \rho v$

Vorsicht: in der Mathematik schreibt man stattdessen gewöhnlich  $\sigma \cdot \rho$ !

in der Notation  
6.9

$$f \circ g(x) = f(g(x))$$

$$f: U \rightarrow V$$
$$v = f(u)$$

Monoid: Menge aller Relationen zwischen  $U$  und  $U$  mit Verknüpfung  $\cdot$ .

- Einselement:  $\Delta = \{(u, u) \mid u \in U\}$ , identische Relation

Beweis:

$$\begin{aligned} \underline{\rho \cdot \Delta} &= \{ (u, w) \mid \exists v \in V: (u, v) \in \rho \text{ und } (v, w) \in \Delta \} && \text{Definition von } \cdot \\ &= \{ (u, w) \mid u \in U \text{ und } (u, w) \in \rho \} && \text{Definition von } \Delta \\ &= \underline{\rho} \end{aligned}$$

- Assoziativgesetz gilt

Beweis: Übung



## 2.1 Beispiel: Komplexprodukte

10

Sei  $H = (U, \cdot)$  Halbgruppe

**Komplexprodukt** von  $\underline{M}, \underline{N} \subseteq U$ :  $M \cdot N = \{m \cdot n \mid m \in M \text{ und } n \in N\}$

Potenzmenge  $\mathcal{P}(U) = \{M \mid M \subseteq U\}$  mit Komplexprodukt ist Halbgruppe

**Beweis:**

$$\begin{aligned} \underline{(M \cdot N)} \cdot Q &= \{x \cdot q \mid x \in M \cdot N \text{ und } q \in Q\} && \text{(Def. Komplexprodukt)} \\ &= \{(m \cdot n) \cdot q \mid m \in M, n \in N, q \in Q\} && \text{(Def. Komplexprodukt)} \\ &= \{m \cdot (n \cdot q) \mid m \in M, n \in N, q \in Q\} && \text{(H ist Halbgruppe)} \\ &= \{m \cdot y \mid m \in M, y \in N \cdot Q\} \\ &= M \cdot (N \cdot Q) && \text{(wie oben)} \end{aligned}$$



Falls  $H$  ein Monoid mit Eins  $\varepsilon$  ist, dann ist die Potenzmenge  $\mathcal{P}(U)$  mit Komplexprodukt und Eins  $e = \{\varepsilon\}$  ein Monoid.

**Beweis:**

$$\begin{aligned} e \cdot M &= \{ \varepsilon \cdot m \mid m \in M \} && \text{(Definition Komplexprodukt)} \\ &= \{ m \cdot \varepsilon \mid m \in M \} && \text{(H ist Monoid)} \\ &= M \cdot e && \text{(Definition Komplexprodukt)} \\ &= H \end{aligned}$$



Sei  $H = (U, \cdot)$  Halbgruppe

**Unterhalbgruppe** von  $H$ : Halbgruppe  $H' = (U', \cdot)$  wobei

- $U' \subseteq U$
- $U'$  algebraisch abgeschlossen bzgl.  $\cdot$       $u, v \in U' \Rightarrow u \cdot v \in U'$

Sei  $M = (U, \cdot, e)$  Monoid

**Untermonoid** von  $M$ :  $M' = (U', \cdot, e)$  wobei

- $(U', \cdot)$  Unterhalbgruppe von  $(U, \cdot)$      ✓
- $e \in U'$

**Beispiel:** Gegeben Monoid  $(\mathbb{N}, \cdot)$ .

- Untermonoid  $\{0, 1\} \subseteq \mathbb{N}$  ist gegeben durch:

$U'$

$\cdot$	0	1
0	0	0
1	0	1



## 2.1 Das Monoid $M^*$ $M^+$

**Gegeben:** Halbgruppe  $H = (U, \cdot)$  bzw. Monoid  $(U, \cdot, e)$ .

Für beliebiges  $M \subseteq U$  setzen wir in

### Erweiterung des Komplexprodukts

$$\begin{aligned} M^0 &= \{e\} && \text{nur für Monoide definiert} \\ M^1 &= M \\ M^{i+1} &= M^i \cdot M \\ M^+ &= \bigcup_{i \in \mathbb{N} \setminus \{0\}} M^i = M^1 \cup M^2 \cup \dots \\ M^* &= \bigcup_{i \in \mathbb{N}} M^i = \{e\} \cup M^+ && \text{nur für Monoide definiert} \end{aligned}$$

$(M^+, \cdot)$  ist kleinste Unterhalbgruppe  $(V, \cdot)$  von  $H$  mit  $M \subseteq V$

$(M^*, \cdot, e)$  ist kleinstes Untermonoid  $(V, \cdot, e)$  mit  $M \subseteq V$  (falls  $H$  Monoid).

Menge  $M$  beliebig  $\subseteq V$

Beweis: Übung

$$\begin{aligned} u, v \in M &\Rightarrow u \cdot v \in M^+ \\ u, v \in M^+ &\Rightarrow u \cdot v \in M^+ \\ \text{z.B. } u \in M^m, v \in M^n &\Rightarrow u \cdot v \in M^{m+n} \subseteq M^+ \end{aligned}$$



### Gegeben:

- Menge  $M$ , Operation  $\cdot$ , definiert für Elemente  $x, y \in M$  |  
aber:  $x \cdot y \in M$  wird nicht vorausgesetzt!
- von  $M$  erzeugte Halbgruppe:  $(M^+, \cdot)$  *generated*
- von  $M$  erzeugtes Monoid:  $(M^*, \cdot, e)$ ,  
falls es ein Einselement  $e$  zur Operation  $\cdot$  gibt  
( $e$  muß nicht unbedingt zu  $M$  gehören!)

→  $M$  heißt **Erzeugendensystem**

→ Wenn  $M$  **endlich** ist, dann heißt die Halbgruppe bzw. das Monoid **endlich erzeugt**



### Beispiel:

- Halbgruppe/Monoid der natürlichen Zahlen ist über  $M = \{1\}$  und Addition  $+$  endlich erzeugt.
- Monoid der Texte über Zeichenvorrat  $\Sigma$  ist über  $\Sigma$  und Konkatenation  $\cdot$  endlich erzeugt.  
 $\Sigma^*$  Menge aller Texte einsch.  $\epsilon$   
über  $\Sigma$
- für Mathematiker:  
ein **Ideal**  $I \subseteq \mathbb{Z}$  ist ein bzgl. Multiplikation mit ganzen Zahlen abgeschlossenes Monoid: aus  $x \in I$ ,  $n \in \mathbb{Z}$  folgt  $n \cdot x \in I$ .
  - tiefliegender Satz der Algebra: alle Ideale im Ring der ganzen Zahlen sind endlich erzeugt



**reflexive Relation**  $\rho \subseteq M^2$  : Für alle  $x \in M$  gilt  $x \rho x$

**symmetrische Relation**  $\rho \subseteq M^2$  :

Für alle  $x, y \in M$  gilt: Aus  $x \rho y$  folgt  $y \rho x$

**antisymmetrische Relation**  $\rho \subseteq M^2$  :

Für alle  $x, y \in M$  gilt: Aus  $x \rho y$  und  $y \rho x$  folgt  $x = y$

**transitive Relation**  $\rho \subseteq M^2$  :

Für alle  $x, y, z \in M$  gilt: Aus  $x \rho y$  und  $y \rho z$  folgt  $x \rho z$

**inverse oder transponierte Relation:**  $\rho^T$  zu  $\rho \subseteq M^2$ :

$y \rho^T x$  genau dann, wenn  $x \rho y$

$\rho^T = \rho$  symmetrisch

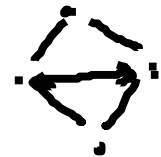
**Äquivalenzrelation:** reflexive, symmetrische, transitive Relation

→ **(Halb-)Ordnung:** reflexive, antisymmetrische, transitive Relation

**Quasiordnung:** reflexive, transitive Relation

**Alternative Ordnung:** für  $x, y \in M$  mit  $x \neq y$  gilt entweder  $x \rho y$  oder  $y \rho x$

→ **Totale oder lineare Ordnung:** alternative Halbordnung



**Zeichenreihen:** Monoid  $\Sigma^*$  über Zeichenvorrat  $\Sigma$  mit Verkettung

**leeres Wort:**

$\varepsilon$  besteht aus null Zeichen und ist neutrales Element von  $\Sigma^*$

**Notationen:**

- $|x|$ : Länge des Wortes  $x$  in Anzahl der Zeichen
- $x^n = \underbrace{x \cdot \dots \cdot x}_{n\text{-mal}}$  : n-fache Wiederholung des Wortes  $x$
- $x^0 = \varepsilon$

**Beobachtung:** Zeichenvorrat  $\Sigma$  ist in der Informatik meist endlich



*Kurt Gödel, 1906 - 1978, bedeutendster Logiker des 20. Jahrhunderts*

Das Monoid  $\Sigma^*$  über endlichem Zeichenvorrat  $\Sigma$  ist abzählbar

**Beweis:**

- Numeriere die Zeichen von  $\Sigma$  mit  $1, \dots, k$   
 $\Rightarrow$  Jede Zeichenreihe  $x_1 \dots x_n$  entspricht  $n$ -Tupel  $(j_1, \dots, j_n)$ ,  $j_j$  ist der Index von  $x_j$  in  $\Sigma$

- Sei  $p_1, p_2, \dots$  die abzählbar unendliche Folge der Primzahlen

- Definiere  $f : \Sigma^* \rightarrow \mathbb{N}$  durch:

$$f(\varepsilon) = 1$$

$$f(x_1, \dots, x_n) = p_1^{j_1} \dots p_n^{j_n}$$

$$\begin{array}{ccccccc} x_1 & & \dots & & x_n & & \\ j_1 & & j_2 & & j_n & & \\ \vdots & & \vdots & & \vdots & & \\ p_1^{j_1} & & p_2^{j_2} & & \dots & & p_n^{j_n} \end{array}$$

- $f$  ist injektiv, weil sich jede natürliche Zahl  $n$  eindeutig in ihre Primzahlen zerlegen lässt, d.h.  $n = p_1^{j_1} \dots p_n^{j_n}$

$\Rightarrow \Sigma^*$  ist abzählbar



**Liste** über Menge  $U$  (auch Sequenz oder Folge): Monoid  $U$  mit Listenkonkatenation ++ und leerer Liste [ ] als neutralem Element

**Notationen:**

- [a<sub>1</sub>, ..., a<sub>n</sub>]: Liste der Elemente a<sub>1</sub>, ..., a<sub>n</sub> (in dieser Reihenfolge)
- append(l, l') = l ++ l' Funktionsschreibweise
- [U]: alternative Notation für U\*  $a_1 a_2 a_3 \dots a_n$

**Beispiel:** [a, b, c] ++ [d, e] = [a, b, c, d, e]

**Beobachtungen:**

- append ist assoziativ
- Unterschied zu Mengen: [1, 2] ≠ [2, 1], [1, 1, 2] ≠ [1, 2]

$$\{1,2\} = \{2,1\} \quad \{1,1,1,1,2,2\} = \{1,2\}$$

Listen spielen in der Informatik eine wichtige Rolle



**Gegeben:** Monoid  $\mathcal{M} = (M, \cdot, e)$

**Inverses Element**  $x'$  zu  $x \in M$ :  $x \cdot x' = e = x' \cdot x$

**Beispiel:** Im Monoid der Funktionen  $f : M \rightarrow M$  mit Verknüpfung besitzen nur die bijektiven Funktionen ein Inverses

**Gruppe:** Monoid, in dem jedes Element ein inverses Element besitzt

**Aber:** In  $\Sigma^*$  und  $[U]$  besitzt nur  $e$  ein inverses Element, also keine Gruppe.

Allerdings gelten:

▪ **Linkskürzungsregel:**

Für alle  $x, y, w \in \Sigma^*$  gilt: Aus  $w \cdot x = w \cdot y$  folgt  $x = y$

*$a b c \cdot \text{inv} = \varepsilon ?$      $\text{inv}$  existiert nicht*

▪ **Rechtskürzungsregel:**

Für alle  $x, y, w \in \Sigma^*$  gilt: Aus  $x \cdot w = y \cdot w$  folgt  $x = y$

$\Rightarrow a_1 \dots a_n = b_1 \dots b_m$  genau dann, wenn  $n = m$  und  $a_i = b_i, i = 1 \dots n$

**Beobachtung:** Obige Kürzungsregeln gelten nicht in allen Monoiden



Gegen sei eine (homogene) Relation:  $\rho \subseteq U \times U$

$x \rho y$

**Gerichteter Graph (Digraph):**  $G = (E, K)$

- endliche oder abzählbare Grundmenge  $E$  von **Ecken** (engl. *vertex*)
- homogene Relation  $K \subseteq E \times E$  **Kanten** (engl. *edge*)  
⇒ gerichtete Graphen und Relationen entsprechen einander

**Notation:**  $e \rightarrow_G e'$  anstelle von  $(e, e') \in K$

$e \rho e'$

- Index  $G$  wird weglassen, wenn klar aus Kontext

$e \rightarrow e'$

endlicher gerichteter Graph:

gerichteter Graph  $G = (E, K)$  mit endlicher Grundmenge

$e - e'$

**ungerichteter Graph:**  $G = (E, K)$  mit symmetrischem  $K$

- die Kanten werden dann ohne Pfeil gezeichnet

$(e, e') \in K \approx$   
 $(e', e) \in K$



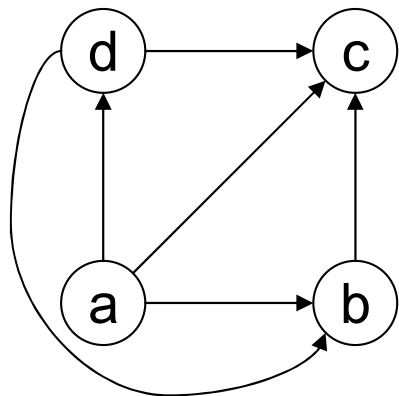
### Beobachtung:

- endliche Graphen können visuell dargestellt werden
- Ecken sind Punkte, Pfeil von  $e$  nach  $e'$  wenn  $e \rightarrow e'$

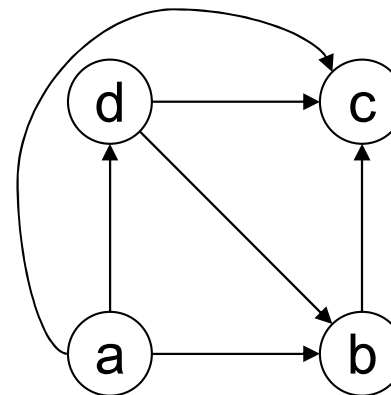
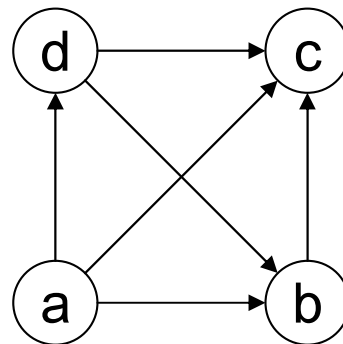
**Beispiel:** Betrachte Graph zur Relation:

$\{a \rightarrow b, a \rightarrow c, a \rightarrow d, b \rightarrow c, d \rightarrow b, d \rightarrow c\}$

$$E = \{a, b, c, d\}$$



3 × gleiche Graph

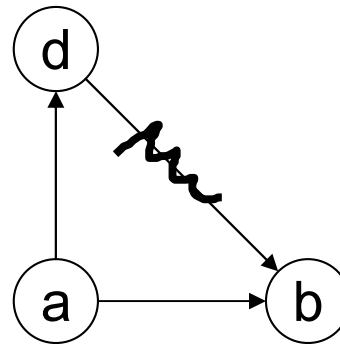
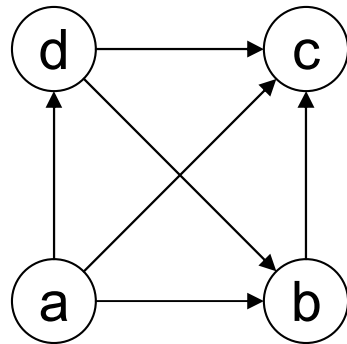


**Teilgraph** eines Graphen  $G = (E, K)$ :

Graph  $G' = (E', K')$  mit  $E' \subset E$  und  $K' = \{(e, e') \in K \mid e, e' \in E'\}$

- $G'$  heißt auch durch  $E'$  induzierter Teilgraph von  $G$

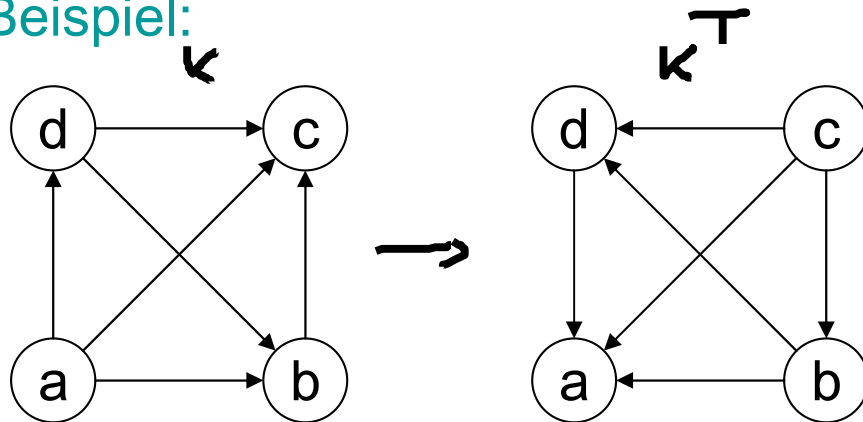
Beispiel:



$$E' = \{a, b, d\}$$

**Dualer Graph** von  $G = (E, K)$ :  $G^T = (E, K^T)$  mit  $K^T = \{(e', e) \mid (e, e') \in K\}$

Beispiel:



$$G = G^T \neq G \text{ ungerichtet}$$



## 2.2 Ecken von Graphen

27

Sei  $G = (E, K)$  gerichteter Graph

**Eingangskanten** von  $e \in E$ :  $\bullet e = \{(e', e) \mid (e', e) \in K\}$

**Ausgangskanten** von  $e \in E$ :  $e\bullet = \{(e, e') \mid (e, e') \in K\}$

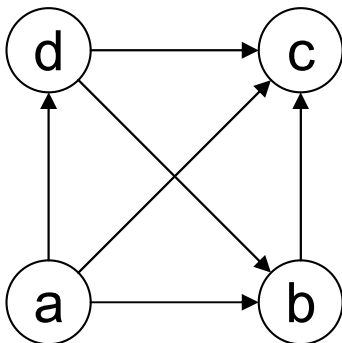
**Eingangsgrad** von  $e \in E$ :  $|\bullet e|$

**Ausgangsgrad** von  $e \in E$ :  $|e\bullet|$

**Eigenschaft:**  $\sum_{e \in E} |\bullet e| = \sum_{e \in E} |e\bullet| = \text{Anzahl Kanten } |K|$

Bei ungerichteten Graphen: **grad(e)** =  $|\bullet e| = |e\bullet|$

### Beispiel



$\bullet a = \emptyset$	$a\bullet = \{a \rightarrow b, a \rightarrow c, a \rightarrow d\}$
$\bullet b = \{a \rightarrow b, d \rightarrow b\}$	$b\bullet = \{b \rightarrow c\}$
$\bullet c = \{a \rightarrow c, b \rightarrow c, d \rightarrow c\}$	$c\bullet = \emptyset$
$\bullet d = \{a \rightarrow d\}$	$d\bullet = \{d \rightarrow b, d \rightarrow c\}$

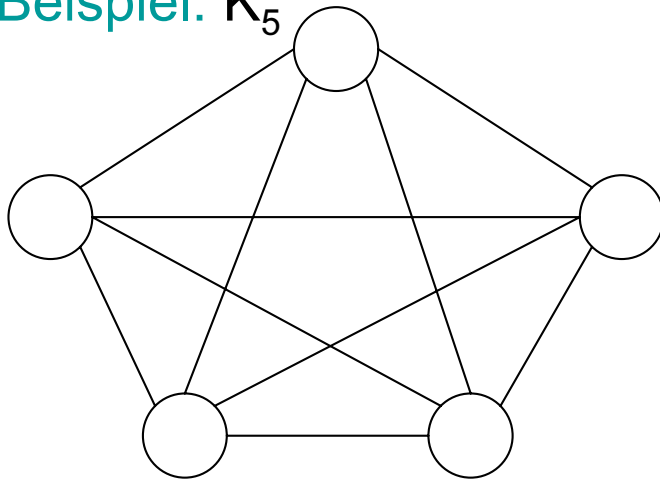


### Vollständiger Graph:

ungerichteter Graph, in dem je zwei Ecken durch eine Kante verbunden sind

**Notation:**  $K_n$  ist vollständiger Graph mit  $n$  Ecken

Beispiel:  $K_5$



$$\frac{n(n-1)}{2}$$

**Übung:** Wieviele Kanten hat ein vollständiger Graph mit  $n$  Ecken?



Sei  $G = (E, K)$  gerichteter Graph

**Weg** in  $G$  der Länge  $n$ : Folge von Ecken  $(e_0, \dots, e_n)$ , so daß  $(e_i, e_{i+1}) \in K$ ,  
 $i = 0, \dots, n-1$

$$e_0 \rightarrow e_1 \rightarrow e_2 \rightarrow \dots \rightarrow e_n$$

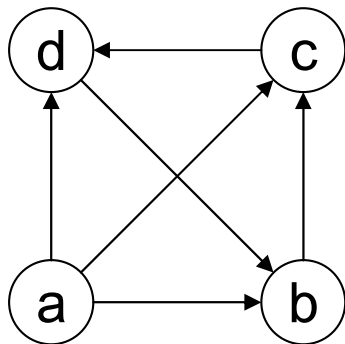
- $e_n$  ist **erreichbar** von  $e_0$
- Definition gilt auch für ungerichtete Graphen



**Zyklus** in  $G$ :  $e_n = e_0$

- in ungerichteten Graphen spricht man von einem **Kreis**

Beispiel:



- Wege von a nach c:
- (a,c)
  - (a,d,b,c)
  - (a,b,c)

Zyklen: (b,c,d,b) (einfach)



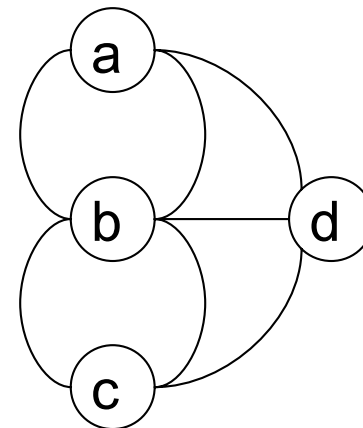
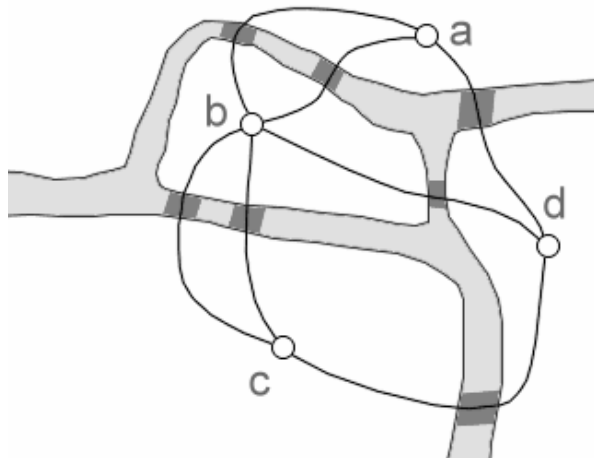
## 2.2 Eulerscher Kreis

30

*Leonhard Euler, 1707 - 1783, schweizer Mathematiker*

**Eulerscher Kreis:** Kreis, der alle Kanten des Graphs genau einmal enthält

**Beispiel:** Königsberger Brückenproblem (Euler, 1736)



- Gibt es einen Weg, der genau einmal über alle sieben Brücken führt und zum Ausgangspunkt zurückführt?

$G = (E, K)$  enthält genau dann eulerschen Kreis, wenn  $\text{grad}(e)$  gerade für alle Ecken  $e \in E$

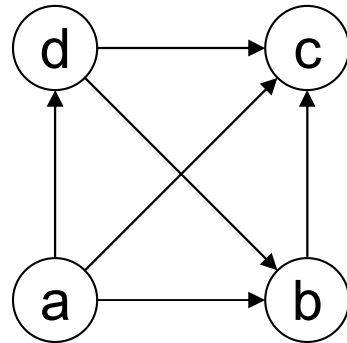
$\Rightarrow$  Königsberger Brückenproblem nicht lösbar

*G zusammenhängend*



**gerichteter azyklischer Graph:** gerichteter Graph ohne Zyklus

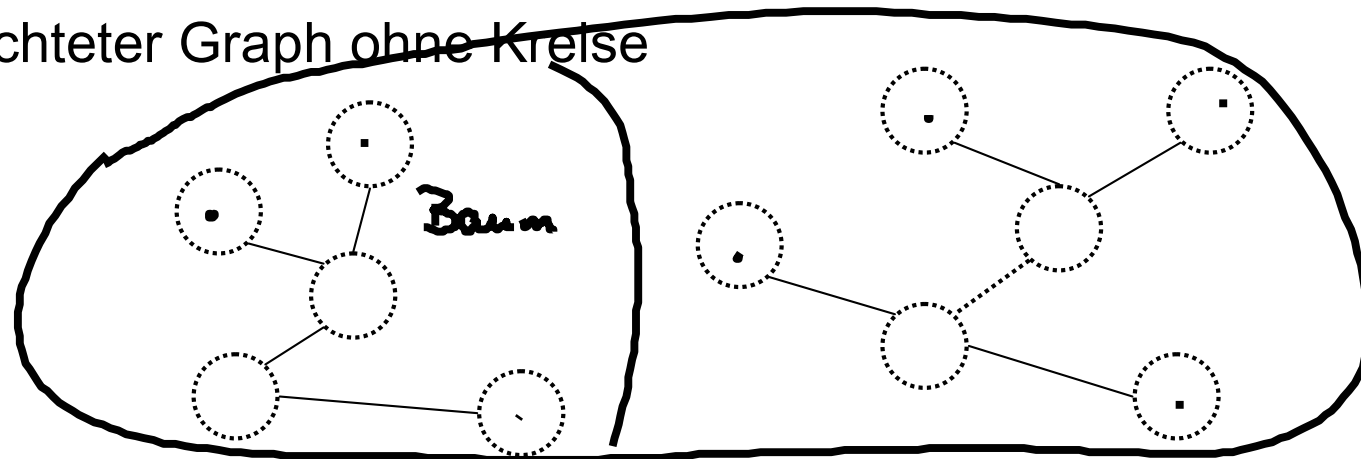
Beispiel:



directed acyclic graph  
DAG

**Wald:** ungerichteter Graph ohne Kreise

Beispiel:



**(ungerichteter) Baum:** Wald, in dem zwei Ecken durch einen Weg verbunden sind

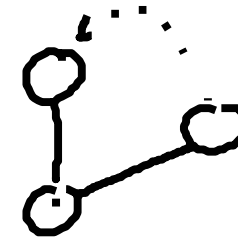
**Blatt eines Waldes:** Ecke mit  $\text{grad}(e) = 1$

▪ Satz: Sei  $G = (E, K)$  ungerichteter Baum. Dann ist  $|E| = |K| + 1$ .

▪ **Beweis:** Induktion über  $|E|$

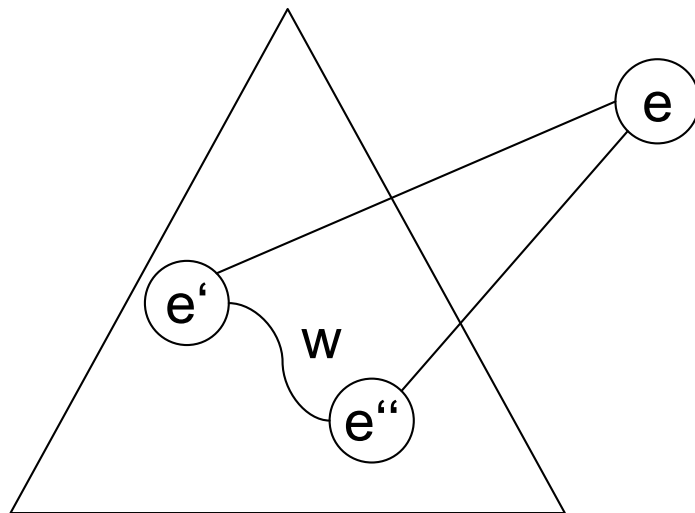
▪ **Induktionsanfang:**

▪  $|E| = 1$ :  Also  $|K| = 0$



▪ **Induktionshypothese:** Aussage gelte für  $|E| = n$

▪ **Induktionsschritt:** Zu zeigen: Aussage gilt für  $|E| = n + 1$



▪ füge neue Ecke  $e$  zu Baum  $G$  mit  $n$  Ecken hinzu

▪ damit Baum entsteht, existiert eine Kante von  $e$  zu Ecke  $e'$  von  $G$

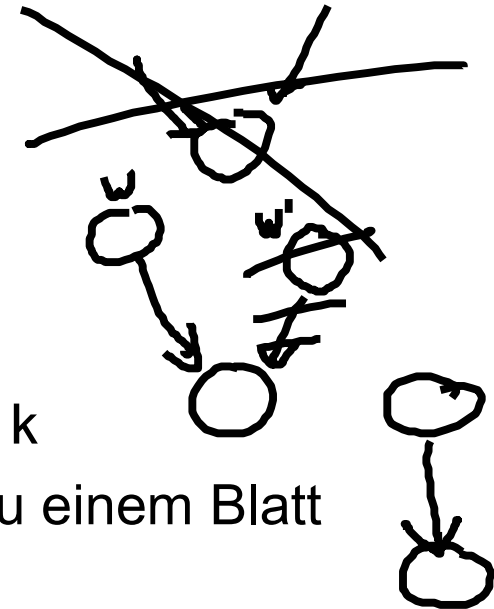
▪ Annahme: es gibt eine weitere solche Ecke  $e''$

▪ da  $G$  Baum gibt es Weg  $w$  von  $e'$  nach  $e''$

⇒ Kreis  $(e, e', \dots, e'', e)$ . **Widerspruch!**

**(gerichteter) Wald:** gerichteter azyklischer Graph  $G = (E, K)$  mit  $|\bullet e| \leq 1$  für alle  $e \in E$

Beispiel:



**Wurzel:** Ecke mit  $|\bullet e| = 0$

Beispiel: a und f

**Baum:** Wald mit genau einer Wurzel

**Blatt:** Ecke mit  $e^\bullet = 0$

Beispiel: e, c, h, j und k

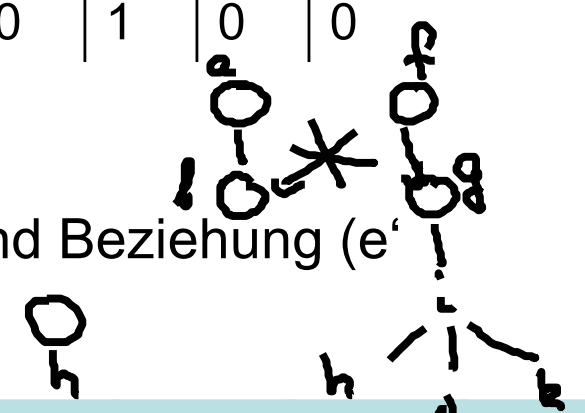
**Höhe von  $e \in E$ :** maximale Länge eines Wegs von  $e$  zu einem Blatt

Beispiel:

Ecke	<u>a</u>	b	c	d	e	<u>f</u>	g	h	i	j	k
Höhe	3	2	0	1	0	3	2	0	1	0	0

**Höhe eines Baums:** Höhe seiner Wurzel

**Eltern und Kinder:** Kante  $e \rightarrow e'$  definiert Eltern-Kind Beziehung ( $e'$  Kind von  $e$ )

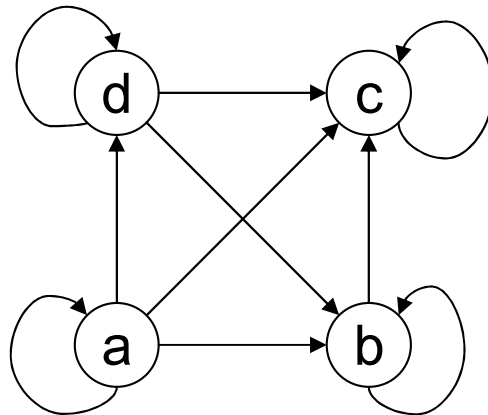


Sei  $G = (E, K)$  gerichteter Graph

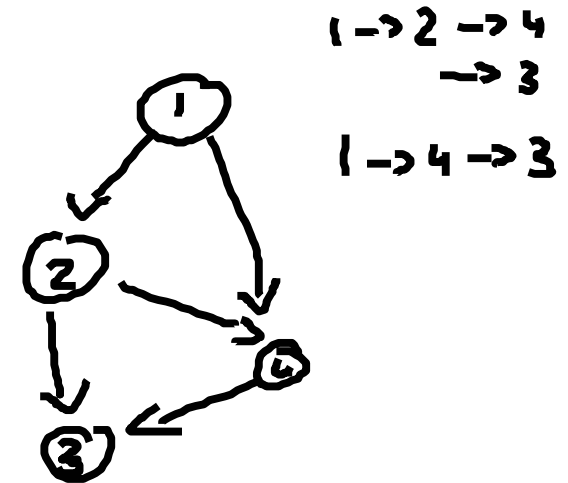
**Schlinge:** Zyklus der Länge 1

**Beobachtung:** Relation ist genau dann reflexiv, wenn jede Ecke eine Schlinge besitzt

**Beispiel:**



$e \rightarrow e$



**Semi-Weg:** Folge von Ecken mit  $(e_0, e_1, \dots, e_n)$ ,  $e_i \rightarrow e_{i+1}$  oder  $e_{i+1} \rightarrow e_i$

**Beispiel:** (c, a, b, d)

**spannender Weg:** Weg, der alle Ecken enthält



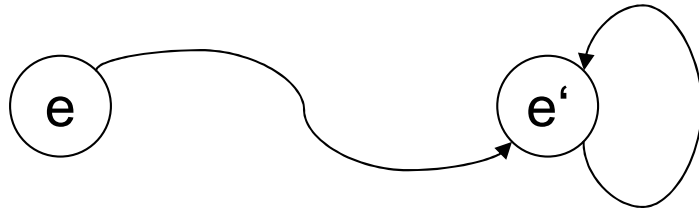
Gegeben: endliche Relation über E. Dann gibt es ein  $n \in \mathbb{N}$ , so daß  $\rho^* = (\Delta \cup \rho)^n$ , wobei  $\Delta = \{(e,e) \mid e \in E\}$  die identische Relation ist

$(\Delta \cup \rho) \cdot (\Delta \cup \rho)$  Aufwand  $n^3$

und  $n = |E| - 1$

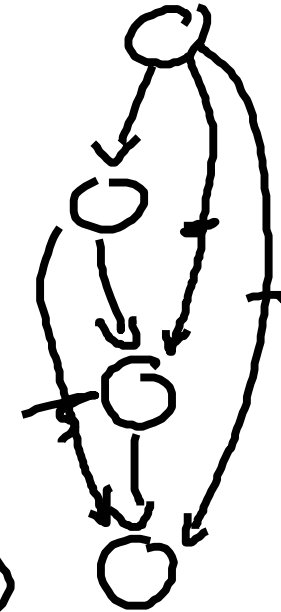
▪ **Beweis:**

▪ Es gilt  $(\Delta \cup \rho)^i \subseteq (\Delta \cup \rho)^n$  für alle  $0 \leq i \leq n$ :



Weg w der Länge i

Weg  $(w, e', \dots, e')$   
(n-i)-mal



$$\Rightarrow (\Delta \cup \rho)^n = \bigcup_{i=0}^n (\Delta \cup \rho)^i$$

$(e_0, e_1, \dots, e_n, e_{n+1}, \dots)$

▪ Sei  $n+1 = |E|$

$\Rightarrow$  Wege der Länge  $\geq n+1$  enthalten wenigstens eine Ecke mehrfach

$\Rightarrow$  jede von  $e \in E$  aus erreichbare Ecke ist mit Weg  $\leq n$  erreichbar

$$\Rightarrow \rho^* = \bigcup_{i=0}^n (\Delta \cup \rho)^i = (\Delta \cup \rho)^n$$

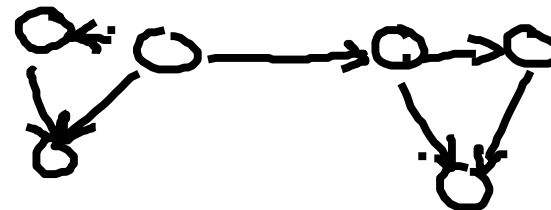
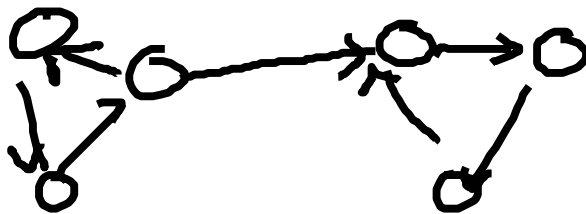


**zusammenhängender Graph:** ungerichteter Graph  $G = (E, K)$ , in dem jedes  $e \in E$  von jedem  $e' \in E$  aus erreichbar ist

**stark zusammenhängender Graph:** gerichteter Graph  $G = (E, K)$ , in dem jedes  $e \in E$  von jedem  $e' \in E$  aus erreichbar ist

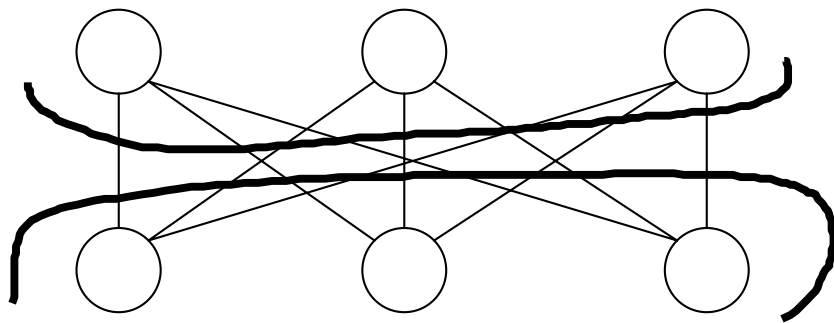
**einseitig zusammenhängender Graph:** gerichteter Graph  $G = (E, K)$ , in dem von je zwei Ecken  $e, e' \in E$  mindestens eine von der anderen erreichbar ist

**schwach zusammenhängender Graph:** gerichteter Graph  $G = (E, K)$ , in dem zwischen je zwei Ecken  $e, e' \in E$  ein Semiweg existiert

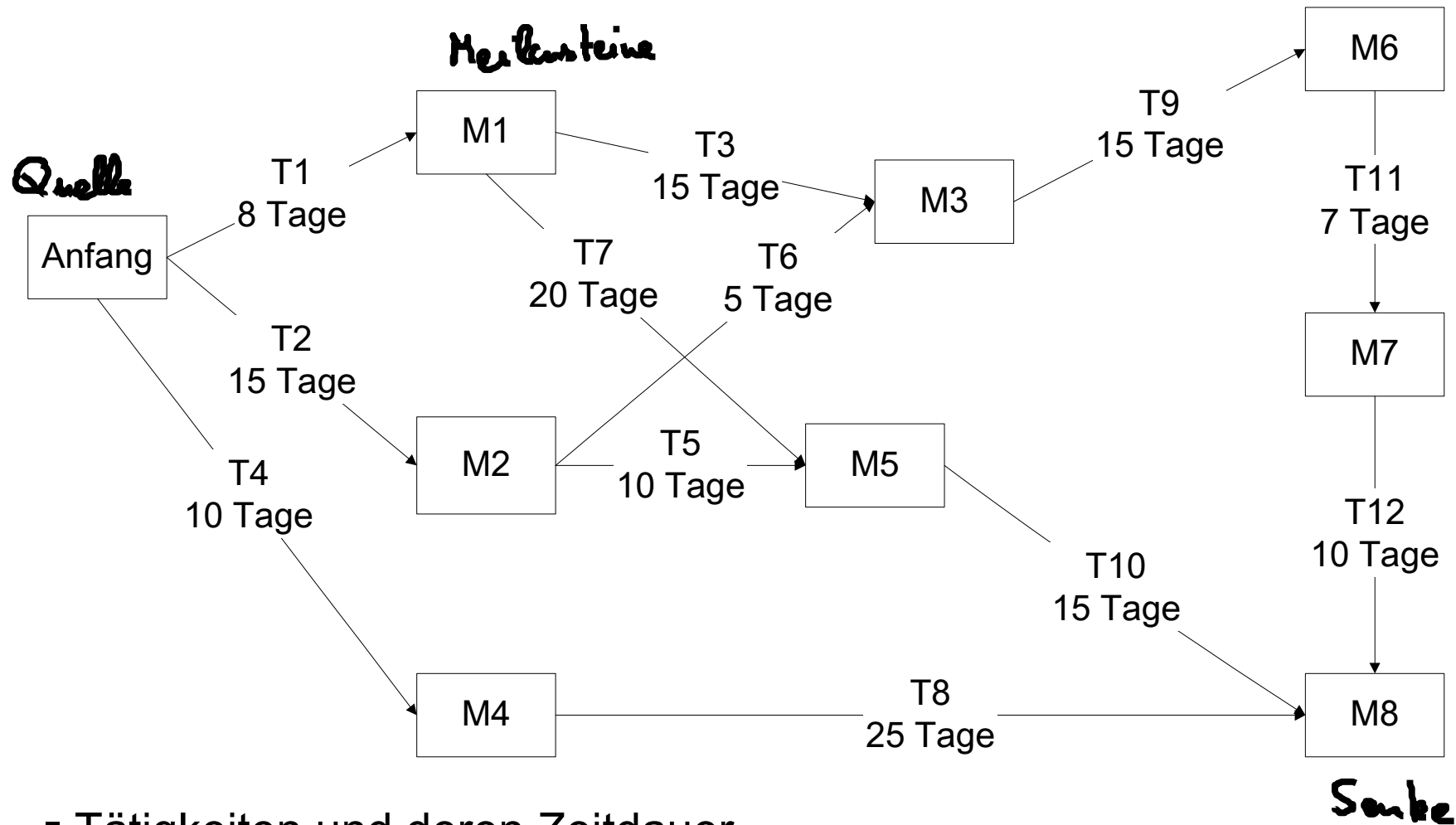


**Bipartiter (paarer) Graph**  $G = (E, K)$ :  $E = U \cup V$  läßt sich in zwei **disjunkte** Mengen  $U, V$  zerlegen, so daß  $K \subseteq U \times V \cup V \times U$

**Beispiel:** vollständig bipartiter (paarer) Graph  $K_{3,3}$



**Anwendungen:** Modellierung von Zuordnungsproblemen

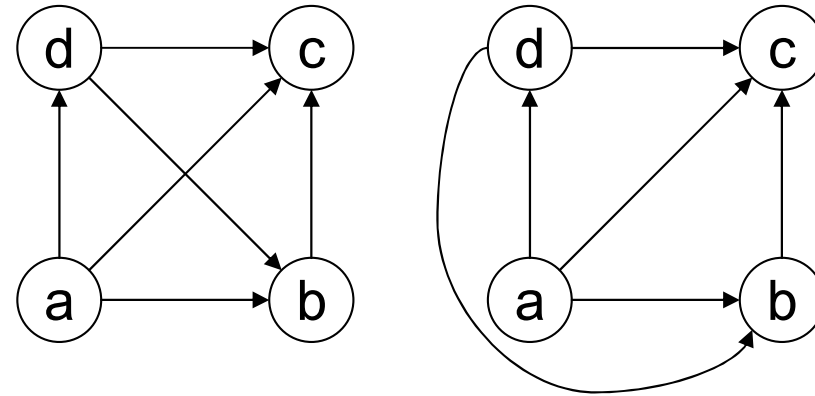


- Tätigkeiten und deren Zeitdauer
- Meilenstein: Beendigung einer Menge von Tätigkeiten
- Zeitpunkt für Fertigstellung eines Meilensteins

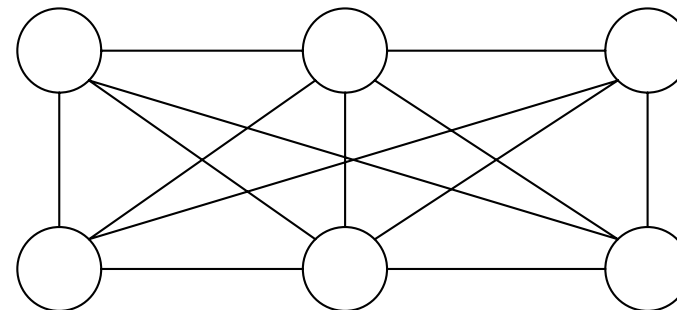


**planarer Graph (ebener Graph)** :  $G = (E, K)$  kann in der Ebene  $\mathbb{R}^2$  so gezeichnet werden, daß alle Ecken auf verschiedene Punkte abgebildet werden und sich keine Kanten kreuzen.

**Beispiel:** planarer Graph



**Beispiel:** nichtplanarer Graph  
enthält  $K_5$  oder  $K_{3,3}$



- Anwendungen:
  - Zeichnen von Graphen
  - elektronische Schaltungen (kreuzungsfreie Leitungen)

**Gegeben:** gerichteter Graph  $G = (E, K)$  mit bijektiver Eckenmarkierung

$$M_E: E \rightarrow \{0, \dots, |E| - 1\}$$

$\Rightarrow$  Ecken  $0, \dots, n - 1$

**Adjazenzmatrix:**  $A = (a_{i,j}) \in \{0, 1\}^{n \times n}$  mit

$$a_{ik} := \begin{cases} 1 & \text{wenn } (i,j) \in K \\ 0 & \text{sonst} \end{cases}$$

$$\begin{array}{c} \begin{matrix} & a & b & c & d \\ a & 0 & 1 & 0 & 1 \\ b & & & & \\ c & & & & \\ \vdots & & & & \\ \vdots & & & & \end{matrix} \end{array}$$

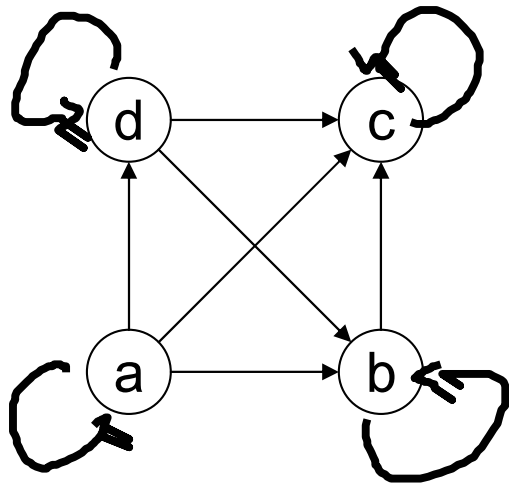
$\Rightarrow$  Darstellung von  $G$  im Binärcode

**Adjazenzlisten:**

Für jede Ecke  $e$ : Darstellung der Menge  $\{e': (e, e') \in K\}$  als Liste

$$a: (b, d, \dots)$$





Adjazenzmatrix:

1	1	1	1
0	1	1	0
0	0	1	0
0	1	1	1

Adjazenzliste:

a: [b, c, d]

b: [c]

c: ~~[d]~~ []

d: [b, c]

- Adjazenzlisten anstelle dünn besetzter Matrizen
- Adjazenzlisten werden zu k-Tupel bei Ausgangsgrad  $\leq k$
- Adjazenzmatrizen:
  - reflexive Relation:  $a_{ii} = 1$  für  $i = 0, \dots, n$
  - symmetrische Relation: Adjazenzmatrix ist symmetrisch
  - Verknüpfung von Relationen  $\rho, \sigma$  gegeben durch  $A = (a_{ij}), B = (b_{ij})$ :

$i (\rho \circ \sigma) j \iff$  es gibt ein  $k$  mit  $i \rho k$  und  $k \sigma j$

$\iff$  es gibt ein  $k$  mit  $a_{ik} = 1$  und  $b_{kj} = 1$

$\iff \sum_{k=0}^n a_{ik} \cdot b_{kj} \geq 1$

$\implies$  Adjazenzmatrix  $C = (c_{ij})$  von  $\rho \circ \sigma$ .

*Handwritten notes:*  
 $n \geq 2$   
 $a_{ik} \cdot b_{kj} + \dots + a_{i0} \cdot b_{0j}$

$$c_{ij} := \begin{cases} 1 & \text{wenn } \sum_{k=0}^n a_{ik} \cdot b_{kj} \geq 1 \\ 0 & \text{sonst} \end{cases}$$

$$A * B = \left( \sum a_{ik} \cdot b_{kj} \right)$$

- transitive Relation  $\rho$   
 $\implies \rho^2 \subseteq \rho \implies a_{ij} \cdot a_{jk} \leq a_{ik}$



$$A \cup A^2 \cup A^3 \cup \dots \cup A^{(n)}$$

**Gegeben:**

- reflexive Relation  $\rho$  über  $E = \{0, \dots, n-1\}$
- Sei  $\sigma^{(k)} = \{ (i,j) \mid \text{Es gibt einen Weg } i \rightarrow e_1 \rightarrow \dots \rightarrow e_{l-1} \rightarrow j, l \leq k, e_j \in \{0, \dots, k-1\}, 1 \leq j < l \}$

Dann ist  $\sigma^{(0)} = \rho$ ,  $\sigma^{(k+1)} = \underbrace{\{(i,j) \mid i \sigma^{(k)} k \text{ und } k \sigma^{(k)} j\}}_{\sigma^{(k)} \cup \sigma^{(k)}}$  und  $\rho^* = \sigma^{(n)}$

**Beweis:** Induktion

- **k = 0:**  $\sigma^0 = \rho$  trivial
- **Induktionsschritt:** Induktionshypothese:  $\sigma^{(k)}$  umfaßt alle Wege beliebiger Länge  $i \rightarrow e_1 \rightarrow \dots \rightarrow e_{l-1} \rightarrow j$  mit  $\{e_1, \dots, e_{l-1}\} \subseteq \{0, \dots, k-1\}$
- $\Rightarrow \{(i,j) \mid i \sigma^{(k)} k \text{ und } k \sigma^{(k)} j\}$  umfaßt alle Wege  $i \rightarrow e_1 \rightarrow \dots \rightarrow e_{l-1} \rightarrow j$  mit  $\{e_1, \dots, e_l\} \subseteq \{0, \dots, k\}$  über Ecke  $k$
- $\Rightarrow$  Behauptung  $\Rightarrow \rho^* = \rho^n = \sigma^{(n)}$



## 2.2 Floyd Warshall Algorithmus

51

Berechnung der transitiven, reflexiven Hülle einer Relation  $\rho$

**Idee:** Berechne  $\sigma^{(k+1)} = \sigma^{(k)} \cup \{(i,j) \mid i \sigma^k k \text{ und } k \sigma^k j\}$  ausgehend von

$$\sigma^{(0)} = \Delta \cup \rho$$

Für Adjazenzmatrizen  $S^{(k)} = (s^{(k)}_{ij})$  von  $\sigma^{(k)}$  gilt:

$$\blacksquare s^{(k+1)}_{ij} = s^{(k)}_{ij} + s^{(k)}_{ik} \cdot s^{(k)}_{kj}$$

$$c_{ij}'' = \sum_k a_{ik} \cdot b_{kj}$$

**Eingabe:** Adjazenzmatrix  $A$  einer Relation  $\rho$

**Ausgabe:** Adjazenzmatrix  $S$  von  $\rho^*$

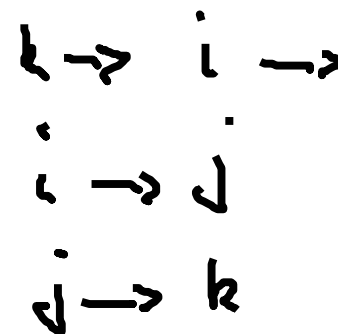
$$S := A;$$

für  $i = 0, \dots, n - 1$  sei  $s_{ii} := 1$

für  $k = 0, \dots, n - 1$

für  $i = 0, \dots, n - 1$

für  $j = 0, \dots, n - 1$  sei  $s_{ij} := s_{ij} + s_{ik} \cdot s_{kj}$



$n^3$  Mult. +  $n^3$  Add



**Beweis:** Betrachte Graph  $G_{\prec} = (U, \prec)$

Sei  $[x] := \{y \in U : x \prec y \prec x\}$

$\Rightarrow [x]$  ist starke Zusammenhangskomponente von  $G_{\prec}$

▪ Definiere  $\mathcal{U} := \{[x] \mid x \in U\}$  und  $f(x) := [x]$

$\Rightarrow \leq$  ist Quasiordnung

▪ Es gelte  $[x] \leq [y]$  und  $[y] \leq [x]$ .

$\Rightarrow$  Für alle  $u \in [x]$ ,  $v \in [y]$  gilt  $u \prec v$  und  $v \prec u$

$\Rightarrow [x] = [y]$

Also ist  $\leq$  Halbordnung



Sei  $\Sigma$  Zeichenvorrat

- $u \in \Sigma^*$  **Präfix** von  $w \in \Sigma^*$  : es gibt  $v \in \Sigma^*$  mit  $\underline{w} = \underline{uv}$

*v Präfix*

**Beispiel:** „Zimmer“ ist Präfix von „Zimmermann“

Relation „ist Präfix“ ist Halbordnung

Gegeben kontextfreie Grammatik  $G = (\Sigma, N, P, Z)$ ,  $V = \Sigma \cup N$

- Ableitungsrelation  $\Rightarrow^* \subseteq V \times V$  ist Quasiordnung
- Wenn  $G$  anständig ist, dann ist  $\Rightarrow^*$  Halbordnung

*Handwritten notes:*  
 $u \leq w$   
 $u \leq v \quad v \leq u$   
 $u \leq uv \quad uv \leq u$   
 $u \leq uv \quad uv \leq v$   
 $v \leq uv \quad uv \leq u$   
 $v \leq uv \quad uv \leq v$   
 $\bigcup_{v' \in \Sigma^*} v' = \Sigma^*$

Markov-Algorithmus über  $\Sigma^*$  (einschließlich Schiffchen)

- Ableitungsrelation  $\Rightarrow^*$  ist Quasiordnung
- Wenn der Algorithmus für alle  $w \in \Sigma^*$  terminiert, dann ist  $\Rightarrow^*$  Halbordnung

*Handwritten notes:*  
 $u \leq v \leq w$   
 $w = uv' = uu'v'$   
 $B \xRightarrow{*} X \xRightarrow{*} B$   
 $\Rightarrow X \quad \text{und} \quad B \neq X$

$$A \xRightarrow{*} B \xRightarrow{*} A$$

$$\Rightarrow X \quad \text{und} \quad B \neq X$$



### Totalordnung $\leq$ auf $U$ :

Halbordnung  $\leq$ , so daß  $x \leq y$  oder  $y \leq x$  für alle  $x, y \in U$

Beispiel:  $\mathbb{N}$  mit  $\leq$

### Kette $K \subseteq U$ :

$K = \{x_1, \dots, x_n\}$  alle  $F.R.$ , unversch., außer  $x_1, x_2$

total geordnete Teilmenge einer halbgeordneten Menge  $(U, \leq)$

Beispiel:  $\emptyset \subseteq \{1\} \subseteq \{1, 2\} \subseteq \{1, 2, 3\} \subseteq \dots$

Beobachtung: Im Graph  $G_{\leq} = (U, \leq)$  entspricht eine Kette einem Weg

absteigende Kette:  $\dots \leq e_{-n} \leq \dots \leq e_{-1} \leq e_0$

Beispiel:  $\dots \subseteq \{3, 4, \dots\} \subseteq \{2, 3, 4, \dots\} \subseteq \{1, 2, 3, 4, \dots\}$

aufsteigende Kette:  $e_0 \leq e_1 \leq \dots \leq e_n \leq \dots$



Gibt es immer ein Hassediagramm?

$\rho \leq \sigma \leq \rho^+ \leq \sigma^+$

Sei  $\sigma$  endliche strenge Halbordnung auf  $U$ . Dann ist  $\rho := \sigma \setminus \sigma^2$  kleinste Relation mit  $\rho^+ = \sigma^+$  und  $G_\rho = (U, \rho)$  ist Hasse-Diagramm.

**Beweis:** Sei  $X_{pq}$  die Menge aller Wege  $p \sigma p_1 \sigma \dots \sigma p_n \sigma q$

- Kein Weg in  $X_{pq}$  besucht ein  $p' \in U$  zweimal, weil  $\sigma$  azyklisch ist.
- Alle Wege in  $X_{pq}$  sind endlich, weil  $\sigma$  endlich ist.

$\Rightarrow$  Aus endlichem  $p' \sigma q'$  lassen sich nur endlich viele

Wege aus  $X_{pq}$  zusammensetzen

$\Rightarrow X_{pq}$  ist endlich

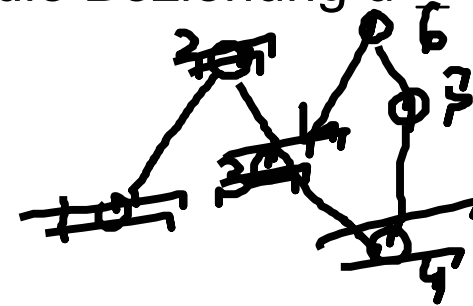


**Gegeben:** endliche Halbordnung  $(U, \prec)$

**Gesucht:** Totalordnung  $(U, \leq)$  so, daß aus  $u \prec v$  die Beziehung  $u \leq v$  folgt.

Topologische Sortierung von  $(U, \prec)$ :

Liste  $L = [e_0, \dots, e_n]$  mit  $e_0 \leq e_1 \leq \dots \leq e_n$



1, 4, 3, 2, 5, 6

4 3 1 5 6 2

**Algorithmus** zur Berechnung einer topologischen Sortierung

- Setze  $L := []$ ;
- Solange  $U \neq \emptyset$  wiederhole
- Wähle eine Ecke  $e$  mit  $|\bullet e| = 0$  und hänge diese hinten an  $L$  an
- Entferne  $e$  aus  $G = (U, \prec)$ ,  
d.h.  $G := (U \setminus \{e\}, \prec \setminus \{(e, e') \mid (e, e') \in \prec\})$

**Gegeben:** Halbordnung  $(U, \leq)$

**fundierte Ordnung:** für jedes nicht leere  $X \subseteq U$  gilt  $\inf(X) \in X$

**Beispiel:**  $(\mathbb{N}, \leq)$ , aber  $(\mathbb{Z}, \leq)$  ist nicht fundiert

**artinsche Ordnung:** (*Emil Artin, 1898-1962, dt. Mathematiker*)  
jede absteigende Kette  $\dots \leq a_{-i} \leq \dots \leq a_{-1} \leq \dots$  bricht ab.

**Beispiel:**  $(\mathbb{N}, \leq)$

**noethersche Ordnung:** (*Emmy Noether, 1882-1935, dt. Mathematikerin*)  
jede aufsteigende Kette  $a_0 \leq a_1 \leq \dots \leq a_i \leq \dots$  bricht ab.

**Beispiel:**  $\Rightarrow^*$  eines terminierenden Semi-Thue-System

**Ausblick:** Diese Eigenschaften spielen eine wichtige Rolle bei Korrektheits- und Terminierungsbeweisen



$(U, \leq)$  ist genau dann fundiert, wenn sie artinsch ist.

**Beweis:**

1. Sei  $(U, \leq)$  fundiert.

Jede Kette  $K$  ist Teilmenge von  $U$

$\Rightarrow \inf K \in K$

2. Sei  $(U, \leq)$  artinsch

**Annahme:** Es gibt  $X \subseteq U$  mit  $\inf(X) \notin X$

Sei  $x_0 \in X$

$\Rightarrow$  es gibt  $x_1 \in X$ , mit  $x_1 < x_0$

$\Rightarrow$  Induktion ergibt nicht abbrechende Kette  $x_0 > x_1 > \dots > x_i > x_{i+1} > \dots$

$\Rightarrow$  Widerspruch!



**Ausgangspunkt:** endliches Alphabet  $(\Sigma, <)$

**Präfixordnung:**  $(\Sigma^*, \sqsubseteq)$  mit  $u \sqsubseteq v$  genau dann, wenn  $u$  Präfix von  $v$  ist

- artinsch, nicht noethersch, fundiert

**lexikographische Ordnung:**  $(\Sigma^*, \prec)$  mit

- $\varepsilon \prec x$  für alle  $x \in \Sigma^*$
- $x \prec y$ , wenn  $x \sqsubseteq y$
- $xay \prec xbz$  für alle  $x, y, z \in \Sigma^*$ ,  $a, b \in \Sigma$  mit  $a < b$
- Die lexikographische Ordnung auf  $\Sigma^*$  ist nicht fundiert, wenn  $\Sigma$  wenigstens zwei Zeichen  $a, b$ ,  $a < b$ , umfaßt:
  - Zu jedem Wort  $a^m b x$ ,  $a \prec a^m b x \prec b$ ,  $x \in \Sigma^*$ , das zumindest ein  $b$  enthält, kann man beliebig viele kleinere Wörter  $a \prec a^m a^n b y \prec a^m b x$  angeben. In einer vorgegebenen (unendlichen) Menge  $X \subseteq \Sigma^*$  mit  $a^m b x \in X$  muß kein kleinstes Element  $\prec a^m b x$  existieren.



### Beweis (Fortsetzung):

- $x_0$  ist kleinster Fixpunkt:

$D = \{f^n(\perp) \mid n \geq 0\} \cup \{x_0\}$  ist kleinste Menge  $M$  mit den Eigenschaften

- $\perp \in M$
- falls  $x \in M$ , dann auch  $f(x) \in M$
- falls  $K \subseteq M$  Kette, so ist auch  $\sup(K) \in M$

**Annahme:** es gibt Fixpunkt  $y \leq x_0$

$\Rightarrow D' = \{f^n(\perp) \mid n \geq 0 \text{ und } f^n(\perp) \leq y\} \cup \{y\}$  erfüllt i., ii. und iii.

$\Rightarrow D \subseteq D'$ , weil  $D$  die kleinste Menge ist, die i., ii. und iii. erfüllt

$\Rightarrow x_0 \leq y$ , also  $x_0 = y$



**Verband**  $(U, \wedge, \vee)$ :  $(U, \wedge)$  und  $(U, \vee)$  sind duale Halbverbände,

- Idempotenzgesetze:  $a \vee a = a$  und  $a \wedge a = a$
- Verschmelzungsgesetze:  $a \vee (a \wedge b) = a$  und  $a \wedge (a \vee b) = a$

### Beobachtungen:

- Es gilt  $a \leq b$  genau dann, wenn  $a \wedge b = a$ .  
Daher ist  $(U, \leq)$  Halbordnung
- Es gilt zusätzlich  $a \leq b$  genau dann, wenn  $a \vee b = b$  ist.
  - Beweis: Sei  $a \leq b$ :  $b = b \vee (b \wedge a) = b \vee a = a \vee b$   
wegen Verschmelzungsgesetz und  $a \wedge b = a$ , Umkehrung analog
  - Schreibweise:  
 $a \vee b = \sup(a, b)$  und  $a \wedge b = \inf(a, b)$

bzw. bei endlicher Grundmenge  $U$ :

$$a \vee b = \max(a, b) \text{ und } a \wedge b = \min(a, b)$$



Buch S. 82  
enthält Fehler



**endlicher Automat:** gedachte Maschine, die Zeichenreihen liest, und über ein beschränktes Gedächtnis des Gelesenen verfügt

- statt Zeichen lesen: jede andere Art unterscheidbarer Eingangssignale verarbeiten

**Zustand:** Gedächtnis über Vorgeschichte der eingelesenen Zeichen

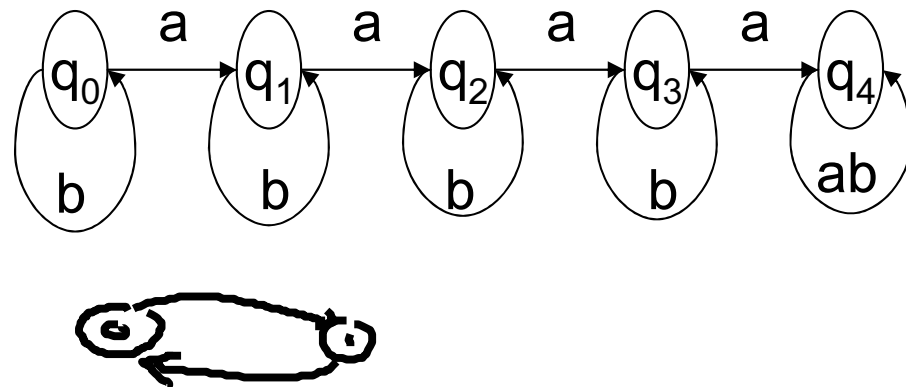
- **endlicher** Automat besitzt **endliche** Menge  $Q$  von Zuständen

**Arbeitsweise:**

- beginnt mit Anfangszustand  $q_0$
- liest Zeichen aus Eingabe und geht in neuen Zustand

⇒ Darstellung als Semi-Thue-System oder gerichteter Graph:

$q_0 a \rightarrow q_1$	$q_0 b \rightarrow q_0$
$q_1 a \rightarrow q_2$	$q_1 b \rightarrow q_1$
$q_2 a \rightarrow q_3$	$q_2 b \rightarrow q_2$
$q_3 a \rightarrow q_4$	$q_3 b \rightarrow q_3$
$q_4 a \rightarrow q_4$	$q_4 b \rightarrow q_4$



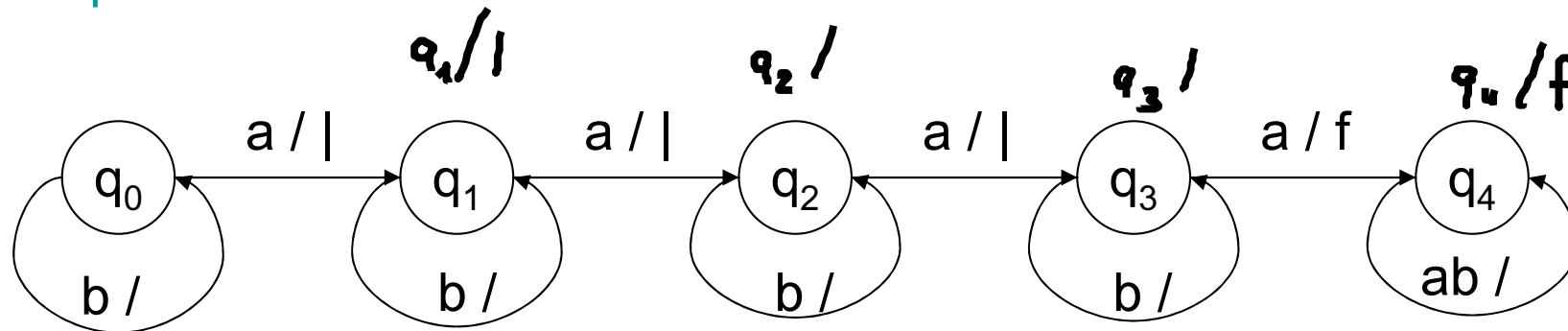
**Klassifikation** nach Art der Ausgabe:

**Mealy-Automat:** Zustandsübergänge erzeugen als Ausgabe Wörter

$t \in \Delta^*$  über Zeichenvorrat  $\Sigma$

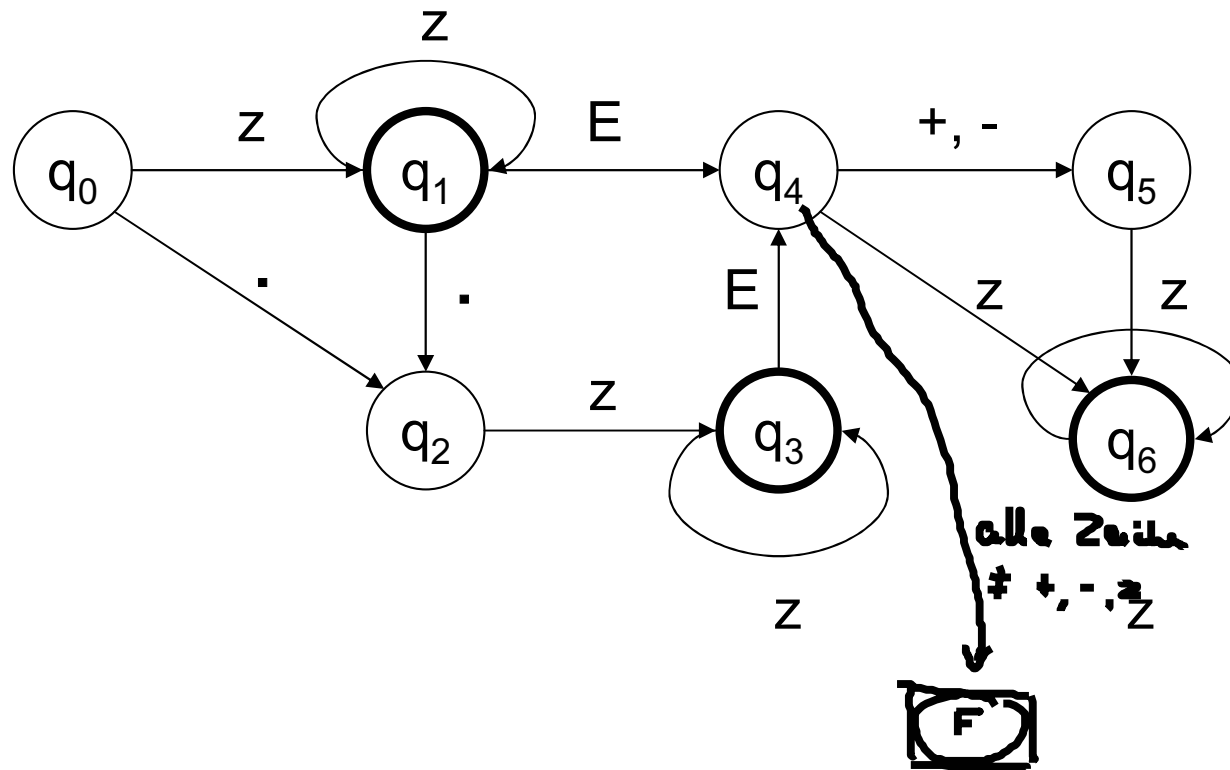
$b \rightarrow \text{---}$   
 $a \rightarrow \text{---}$   
 $a b \rightarrow \text{---}$

Beispiel:



**Moore-Automat:** Zustände erzeugen als Ausgabe Wörter  $t \in \Delta^*$  über Zeichenvorrat  $\Sigma$





- q<sub>0</sub> Startzustand
- q<sub>1</sub> ganze Zahl
- q<sub>2</sub> Punkt
- q<sub>3</sub> Festpunktzahl
- q<sub>4</sub> Exponent
- q<sub>5</sub> Vorz. Exponent
- q<sub>6</sub> Gleitpunktzahl

q<sub>1</sub> akzeptiert natürliche Zahlen, q<sub>3</sub> akzeptiert Festpunktzahlen, q<sub>6</sub> akzeptiert Gleitpunktzahlen



### endlicher Akzeptor:

$A = (\Sigma, Q, q_0, F, R)$  wobei

- $\Sigma$  Zeichenvorrat,
- $Q$  nicht leere Menge von Zuständen ( $Q \cap \Sigma = \emptyset$ ),
- $q_0 \in Q$  Anfangszustand,
- $F \subseteq Q$  Menge der Endzustände und
- $R$  Menge von Zustandsübergangsregeln der Form  $qa \rightarrow q'$  mit  $q, q' \in Q, a \in \Sigma$  ist.

$$q_0 a a a \implies q_1 a a \implies q_2 a \implies q_3$$

**Bemerkung:**  $(\Sigma \cup Q, \rightarrow)$  ist Semi-Thue-System  $q_0 a a a \implies^* q_3$



**Satz:**  $L \subseteq \Sigma^*$  ist  $L = L(A)$  für einen Akzeptor  $A = (\Sigma, Q, q_0, F, R)$ , genau dann wenn  $L = L(G)$  für eine reguläre Grammatik  $G = (\Sigma, N, P, Z)$  ist.

**Beweis:** „nur dann, wenn“. Sei  $G$  reguläre Grammatik

$\Rightarrow$  alle Produktionen haben die Form  $X \rightarrow aB$ ,  $X \rightarrow a$  oder  $X \rightarrow \varepsilon$

- Konstruiere Akzeptor  $A = (\Sigma, Q, q_0, F, R)$  wie folgt:

$$N' = \{N_{Xa} \mid X \rightarrow a \in P\} \cup \{X \mid X \rightarrow \varepsilon \in P\}$$

$$Q = N \cup N'$$

$$q_0 = Z$$

$$F = N'$$

$$R = \{Xa \rightarrow Y \mid X \rightarrow aY \in P\} \cup \{Xa \rightarrow N_{Xa} \mid X \rightarrow a \in P\}$$

$$X \rightarrow a Y$$

$$X \rightarrow a$$

$\downarrow$  Automata

$$X a \rightarrow Y$$

$$X a \rightarrow N_{Xa}$$

- **Behauptung:**

Für alle  $X \in N$  gilt:  $X \Rightarrow_G^* w$  genau dann, wenn

$Xw \Rightarrow_A^* H$  für ein  $H \in F$



**Gegeben:** reguläre Grammatik mit Produktionen

$$Z \rightarrow aX \mid bX$$

$$X \rightarrow aX \mid bX \mid \varepsilon$$

$$L(G) = \{a, b\}^+$$

a	b
aa	ba
ab	bb
aaa	
aab	
aba	
abb	

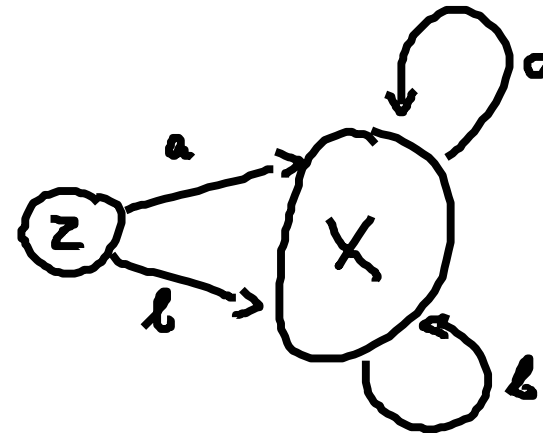
**endlicher Automat:**

Anfangszustand Z

$$Za \rightarrow X \quad Zb \rightarrow X$$

$$Xa \rightarrow X \quad Xb \rightarrow X$$

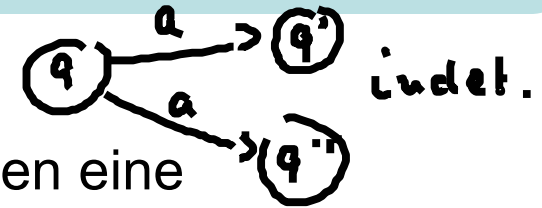
Endzustand: X



▪ Bei bisherigen Beispielen:

- Übergänge  $qa \rightarrow q'$  bilden zusammengenommen eine Übergangsfunktion  $\delta: Q \times \Sigma \rightarrow Q$

- Ein endlicher Automat, der die Eigenschaft besitzt, daß die Übergänge zusammengenommen eine (Übergangs-)Funktion bilden, heißt **deterministisch**



▪ Allgemeiner Fall: Indeterminismus zugelassen

- Die Übergänge definieren ein durch  $P$  endlich erzeugtes Semi-Thue-System mit Eingabezeichenvorrat  $\Sigma$  und syntaktischen Hilfszeichen  $q \in Q$
- Die Auffassung eines endlichen Automaten als Semi-Thue-System stellt die Beziehung zu regulären Chomsky-Grammatiken her

Übergangsmatrix

$\delta$	a	b	c
$q_1$			
$q_2$			$q'$



- Ein regulärer Ausdruck  $R$  über einem Zeichenvorrat  $\Sigma$  ist induktiv definiert durch:

1. Das leere Wort ist ein regulärer Ausdruck

 $\epsilon$ 

2.  $a \in \Sigma$  ist ein regulärer Ausdruck

 $a \quad b \quad c$ 

3. Ist  $R$  ein regulärer Ausdruck, dann auch  $(R)^*$

 $R^* \quad \epsilon \quad R \quad RR$ 

4. Sind  $R, S$  reguläre Ausdrücke, so sind auch  $(RS)$  und  $(R + S)$  reguläre Ausdrücke

 $RS$ 

□  $R + S$  bedeutet „ $R$  oder  $S$ “ (Vereinigung)

- Meist gilt auch die leere Menge  $\emptyset$  als regulärer Ausdruck

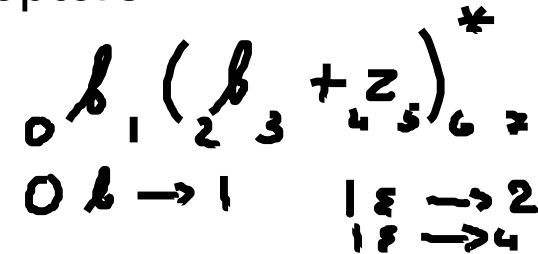
### Beispiele:

- Bezeichner:  $b(b+z)^*$
- Ganze Zahlen:  $zz^*$
- Dezimalbrüche:  $zz^* ' / ' zz^*$



- Beweis des vorhergehenden Satzes liefert ein konstruktives Vorgehen zum Aufbau eines endlichen Akzeptors

**Skizze dieses konstruktiven Vorgehens:**



Geg.: regulärer Ausdruck R

Durch folgende Regeln wird R in einen endlichen Akzeptor überführt:

1. Füge zu Beginn von R sowie nach jedem terminalen Zeichen eine Zahl ein, die einen Zustand repräsentiert
2. Einzelnes Zeichen bedeutet: Zustandsübergang
3. Vereinigung führt zu Zustandsübergängen der zu Beginn gegebenen Zustände in alle durch Einzelzeichen erreichbaren Folgezustände
4. Kleenescher Stern  $S^*$  führt zu Zustandsübergängen aus sämtlichen Endzuständen von S in die aus den Anfangszuständen aus S mit einem Zeichen erreichbaren Zustände

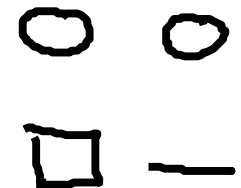


*Carl Adam Petri, geb. 1926, früherer Institutsleiter der GMD, Bonn*

**Petri-Netze:** verallgemeinern endliche Automaten und sind die einfachste Form, parallele und verteilte Systeme zu beschreiben

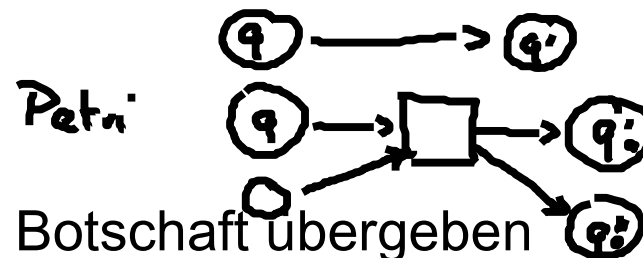
**Grundidee:** Gegeben ein paarer Graph von Übergängen (**Transitionen:** Kästchen) und Zuständen (**Stellen:** Kreise).

- eine Stelle kann mehrere Transitionen als Nachfolger haben
- eine Transition kann mehrere Stellen als Nachfolger haben
- Regeln: jede **Stelle** kann eine oder mehrere Marken enthalten
  - **Feuern** einer Transition: jede Vorgängerstelle gibt eine Marke ab, jede Nachfolgerstelle erhält eine Marke (dabei Veränderung der Markenanzahl möglich)



**Interpretation z.B.:**

- Stelle: Prozeß, Vorgang; Transition: Botschaft übergeben
- Stelle: Zustand; Transition: Zustandsübergang



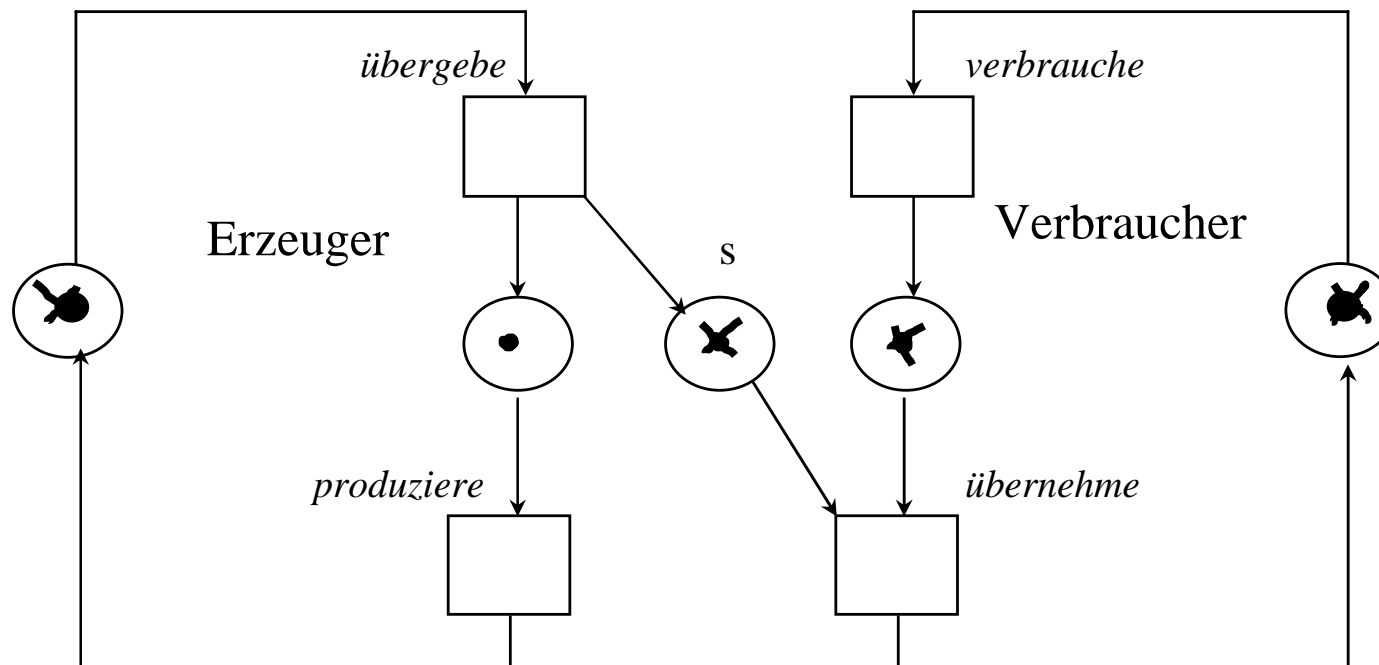
- **Petrinetze** werden in der Informatik als Systemmodelle für Vorgänge und Organisationen verwendet, um Zustandsübergänge und Datenflüsse auch in parallelen und verteilten Systemen zu beschreiben
  
- Bestandteile und Arbeitsweise eines Petrinetzes  $P = (S, T, F)$ 
  - **S: Stellen** repräsentieren Zustände, Bedingungen, Pufferinhalte
  - **T: Transitionen** (Übergänge) repräsentieren Zustandsübergänge, Aktivitäten oder Verarbeitung von Pufferinhalten
  - **F: Kanten** repräsentieren kausale oder zeitliche Abhängigkeiten
    - $s \rightarrow t$  bedeutet: Bedingung  $s \in S$  muß erfüllt sein, damit die Transition  $t \in T$  schaltet (feuert)  
 $s$  heißt **Vorstelle** von  $t$
  
    - $t \rightarrow s$  bedeutet: Zustand  $s \in S$  wird erreicht, wenn die Transition  $t \in T$  feuert  
 $s$  heißt **Nachstelle** von  $t$



## 2.5 Beispiel: Erzeuger-Verbraucher-Problem

120

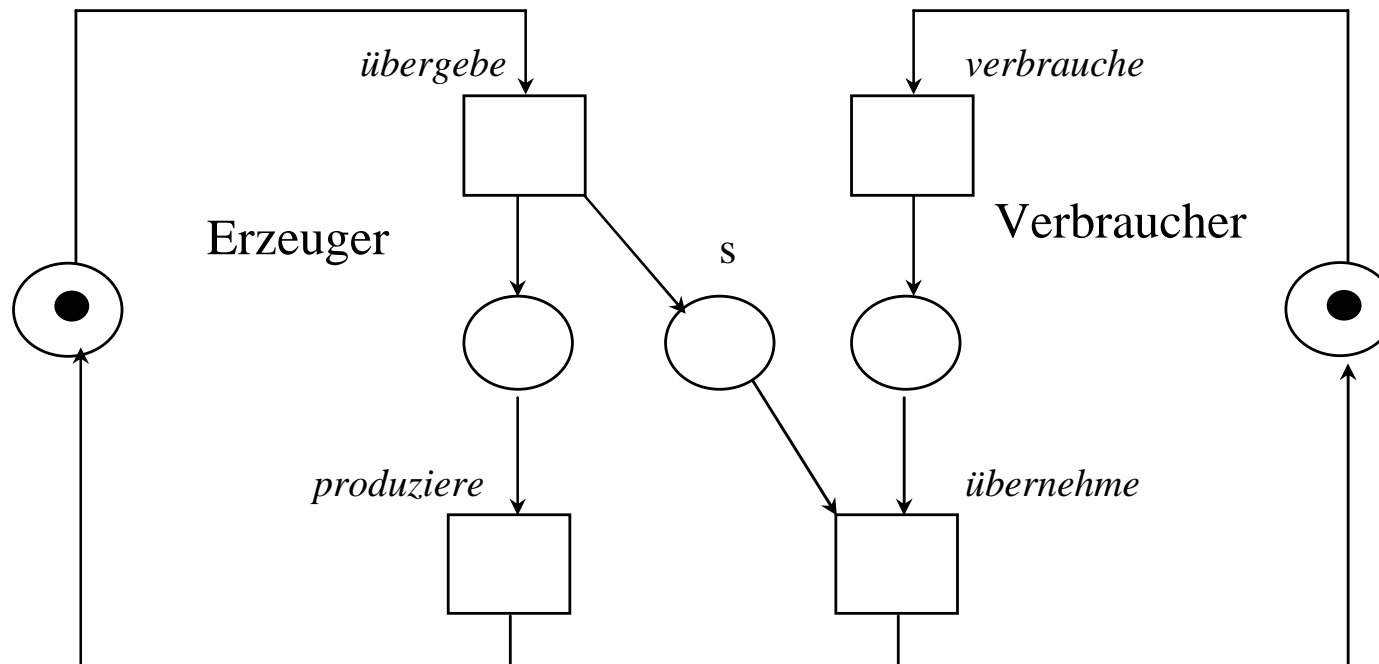
Ziel: Synchronisierung von zwei nebenläufigen Prozessen



*s*: privates Semaphore des Verbrauchers

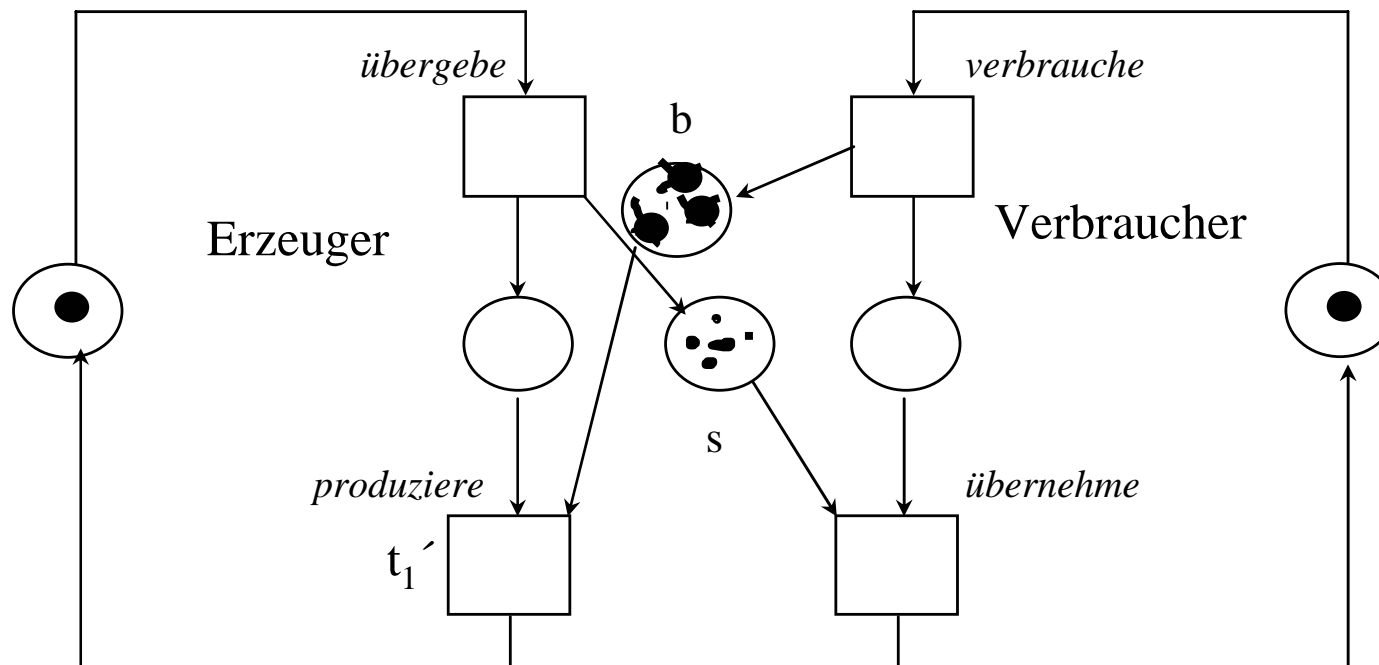


**Ziel:** der Verbraucher darf nur weitermachen, wenn er vom Erzeuger etwas geliefert bekommt



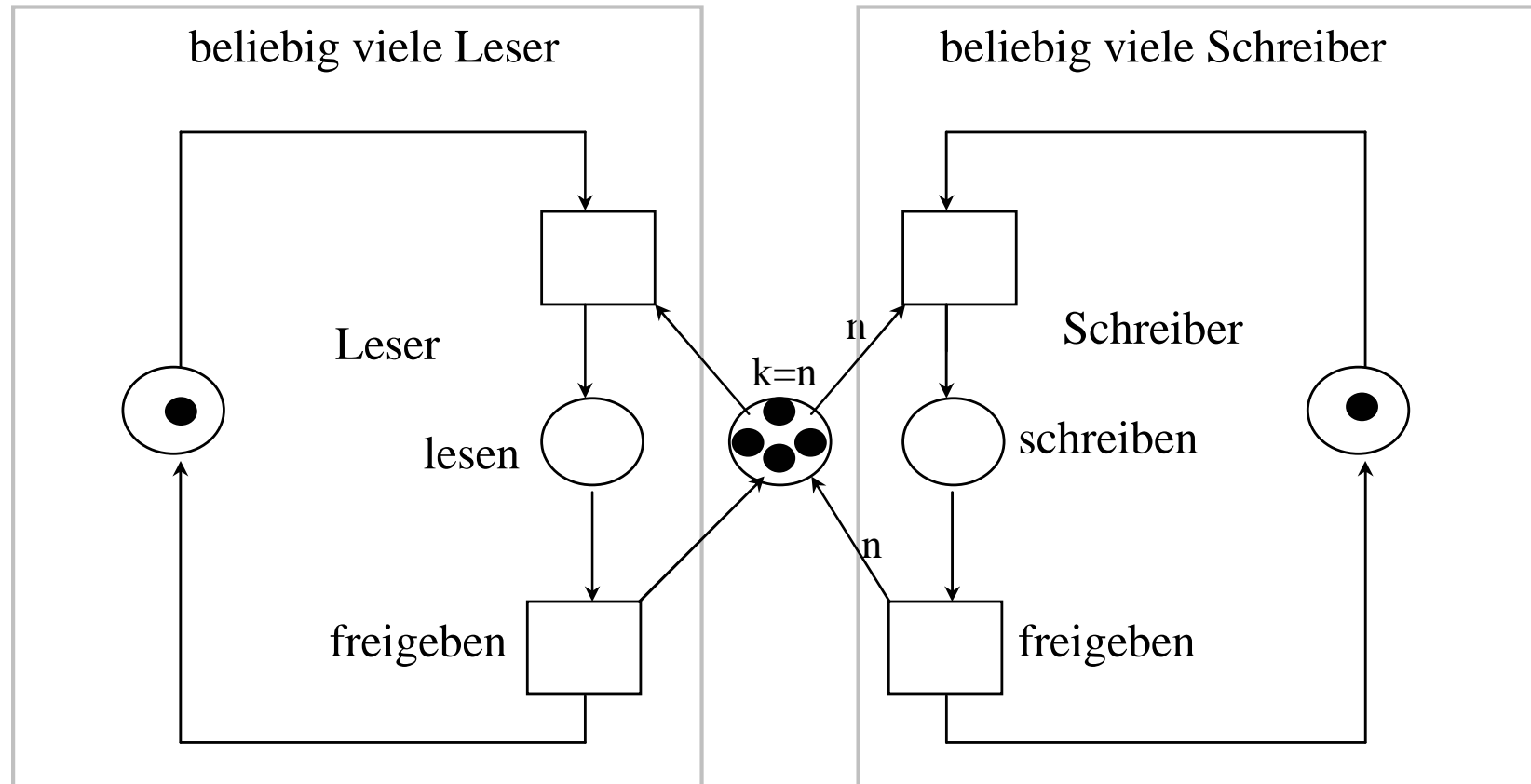
- Die Stelle *s* kann unbeschränkt viele Marken aufnehmen:
  - Erzeuger-Verbraucher mit unbeschränktem Puffer *s*

**Ziel:** der Verbraucher darf nur weitermachen, wenn er vom Erzeuger etwas geliefert bekommt und genügend Platz im Puffer ist



- Wenn die Stelle  $b$   $n$  Marken enthält, kann Übergang  $t_1'$   $n$ -mal feuern, sonst wartet der Erzeuger
  - Hier also  $3+1=4$  Marken in Puffer  $s$  möglich, bevor der Erzeuger stoppt

**Ziel:**  $n$  Leser dürfen gleichzeitig Daten lesen, ein Schreiber verlangt Exklusivzugriff und sperrt alle Leser aus



- $k=n$ : Stelle kann maximal  $n$  Marken aufnehmen
- $n$  an einer Kante:  $n$  Marken fließen gleichzeitig
- keine Fairness !